WHAT IS CLAIMED IS:

1. A method comprising:
selecting a first allocated memory block from a plurality of allocated memory blocks, wherein the first allocated memory block includes a first allocated memory block address;
searching other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block;
verifying that the first allocated memory block is a memory leak when the reference to the first allocated memory block is not found in the other allocated memory blocks of the plurality of allocated memory blocks; and
reporting the first allocated memory block as a memory leak.

2. The method of claim 1 wherein the selecting the first allocated memory block from the plurality of allocated memory blocks further comprises:
selecting the first allocated memory block address from operating system memory management information.

3. The method of claim 1 wherein each of the plurality of allocated memory blocks includes a header portion, and wherein the searching other allocated memory blocks of the plurality of allocated memory blocks further comprises:
searching the header portions of the other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block.

4. The method of claim 1 wherein the searching other allocated memory blocks of the plurality of allocated memory blocks further comprises:
searching for an occurrence of the first allocated memory block address in the other allocated memory blocks of the plurality of allocated memory blocks.

5. The method of claim 1 further comprising:
examining a reference counter corresponding to the first allocated memory block.

6. The method of claim 1 wherein the verifying that the first allocated memory block is a memory leak further comprises:

determining whether the first allocated memory block has been deallocated.

7. The method of claim 1 wherein the first allocated memory block includes a header portion, and wherein the verifying that the first allocated memory block is a memory leak further comprises:

examining the header portion of the first allocated memory block.

8. The method of claim 1 wherein the verifying that the first allocated memory block is a memory leak further comprises:

examining free block memory management information maintained by an operating system.

9. The method of claim 1 wherein the reporting the first allocated memory block as a memory leak further comprises:

displaying to a user at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

10. The method of claim 1 wherein the reporting the first allocated memory block as a memory leak further comprises:

storing in a data structure at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

11. The method of claim 1 further comprising:

searching the first allocated memory block for a reference to at least one of the plurality allocated memory blocks; and

storing the first allocated memory block address in a contingency chain corresponding to the at least one of the plurality allocated memory

blocks when a reference to the at least one of the plurality allocated memory blocks is found in the first allocated memory block.

12. The method of claim 1 further comprising:

examining a contingency chain corresponding to one of the plurality of allocated memory blocks to determine whether any of the plurality of allocated memory blocks references the one of the plurality of allocated memory.

13. The method of claim 1 further comprising:

forming a contingency chain for each of the plurality of allocated memory blocks, wherein each contingency chain is indexed by an allocated memory block address of the corresponding each of the plurality of allocated memory blocks.

14. A system comprising:

a memory;

a processor coupled to the memory; and

a memory leak detection system (MLDS) engine, wherein at least a portion of the MLDS engine is encoded as instructions stored in the memory and executable on the processor, and wherein the MLDS engine is configured to:

select a first allocated memory block from a plurality of allocated memory blocks stored in the memory, wherein the first allocated memory block includes a first allocated memory block address;

search other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block;

verify that the first allocated memory block is a memory leak when the reference to the first allocated memory block is not found in the other allocated memory blocks of the plurality of allocated memory blocks; and

report the first allocated memory block as a memory leak.

15. The system of claim 14 further comprising at least of an MLDS data structure application programming interface (API), an MLDS command API, an MLDS data structure, and a command line interface (CLI) parser stored in at least one of the memory and a storage device accessible by the processor.

16. The system of claim 14 wherein the MLDS engine is further configured to:

select the first allocated memory block address from operating system memory management information.

17. The system of claim 14 wherein each of the plurality of allocated memory blocks includes a header portion, and wherein the MLDS engine is further configured to:

search the header portions of the other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block.

18. The system of claim 14 wherein the MLDS engine is further configured to:

search for an occurrence of the first allocated memory block address in the other allocated memory blocks of the plurality of allocated memory blocks.

19. The system of claim 14 wherein the MLDS engine is further configured to:

examine a reference counter corresponding to the first allocated memory block.

20. The system of claim 14 wherein the MLDS engine is further configured to determine whether the first allocated memory block has been deallocated.

21. The system of claim 14 wherein the first allocated memory block includes a header portion, and wherein the MLDS engine is further configured to:

examine the header portion of the first allocated memory block.

22. The system of claim 14 wherein the MLDS engine is further configured to:

examine free block memory management information maintained by an operating system.

23. The system of claim 14 wherein the MLDS engine is further configured to:

display at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

24. The system of claim 14 wherein the MLDS engine is further configured to:

store in a data structure at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

25. The system of claim 14 wherein the MLDS engine is further configured to:

search the first allocated memory block for a reference to at least one of the plurality allocated memory blocks; and

store the first allocated memory block address in a contingency chain corresponding to the at least one of the plurality allocated memory blocks when a reference to the at least one of the plurality allocated memory blocks is found in the first allocated memory block.

26. The system of claim 14 wherein the MLDS engine is further configured to:

examine a contingency chain corresponding to one of the plurality of allocated memory blocks to determine whether any of the plurality of allocated

memory blocks references the one of the plurality of allocated memory.

27. The system of claim 14 wherein the MLDS engine is further configured to:

form a contingency chain for each of the plurality of allocated memory blocks, wherein each contingency chain is indexed by an allocated memory block address of the corresponding each of the plurality of allocated memory blocks.

28. A computer readable medium comprising program instructions executable on a processor, the computer readable medium being at least one of an electronic storage medium, a magnetic storage medium, an optical storage medium, and a communications medium conveying signals encoding the instructions, wherein the program instructions are operable to implement each of:

selecting a first allocated memory block from a plurality of allocated memory blocks, wherein the first allocated memory block includes a first allocated memory block address;

searching other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block;

verifying that the first allocated memory block is a memory leak when the reference to the first allocated memory block is not found in the other allocated memory blocks of the plurality of allocated memory blocks; and

reporting the first allocated memory block as a memory leak.

29. The computer readable medium of claim 28 wherein the selecting the first allocated memory block from the plurality of allocated memory blocks further comprises:

selecting the first allocated memory block address from operating system memory management information.

30. The computer readable medium of claim 28 wherein each of the plurality of allocated memory blocks includes a header portion, and wherein the searching

other allocated memory blocks of the plurality of allocated memory blocks further comprises:

searching the header portions of the other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block.

31. The computer readable medium of claim 28 wherein the searching other allocated memory blocks of the plurality of allocated memory blocks further comprises:

searching for an occurrence of the first allocated memory block address in the other allocated memory blocks of the plurality of allocated memory blocks.

32. The computer readable medium of claim 28 further comprising program instructions are operable to implement:

examining a reference counter corresponding to the first allocated memory block.

33. The computer readable medium of claim 28 wherein the verifying that the first allocated memory block is a memory leak further comprises:

determining whether the first allocated memory block has been deallocated.

34. The computer readable medium of claim 28 wherein the first allocated memory block includes a header portion, and wherein the verifying that the first allocated memory block is a memory leak further comprises:

examining the header portion of the first allocated memory block.

35. The computer readable medium of claim 28 wherein the verifying that the first allocated memory block is a memory leak further comprises:

examining free block memory management information maintained by an operating system.

36. The computer readable medium of claim 28 wherein the reporting the first allocated memory block as a memory leak further comprises:

displaying to a user at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous

block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

37. The computer readable medium of claim 28 wherein the reporting the first allocated memory block as a memory leak further comprises:

storing in a data structure at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

38. The computer readable medium of claim 28 further comprising program instructions are operable to implement each of:

searching the first allocated memory block for a reference to at least one of the plurality allocated memory blocks; and

storing the first allocated memory block address in a contingency chain corresponding to the at least one of the plurality allocated memory blocks when a reference to the at least one of the plurality allocated memory blocks is found in the first allocated memory block.

39. The computer readable medium of claim 28 further comprising program instructions are operable to implement each of:

examining a contingency chain corresponding to one of the plurality of allocated memory blocks to determine whether any of the plurality of allocated memory blocks references the one of the plurality of allocated memory.

40. The computer readable medium of claim 28 further comprising program instructions are operable to implement:

forming a contingency chain for each of the plurality of allocated memory blocks, wherein each contingency chain is indexed by an allocated memory block address of the corresponding each of the plurality of allocated memory blocks.

41. An apparatus comprising:

a means for selecting a first allocated memory block from a plurality of allocated memory blocks, wherein the first allocated memory block includes a first allocated memory block address;

a means for searching other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block;

a means for verifying that the first allocated memory block is a memory leak when the reference to the first allocated memory block is not found in the other allocated memory blocks of the plurality of allocated memory blocks; and

a means for reporting the first allocated memory block as a memory leak.

42. The apparatus of claim 41 wherein each of the plurality of allocated memory blocks includes a header portion, and wherein the apparatus further comprises:

a means for searching the header portions of the other allocated memory blocks of the plurality of allocated memory blocks for a reference to the first allocated memory block.

43. The apparatus of claim 41 further comprising:

a means for searching for an occurrence of the first allocated memory block address in the other allocated memory blocks of the plurality of allocated memory blocks.

44. The apparatus of claim 41 further comprising:

a means for displaying to a user at least one of: a program counter value, a process identification value, a process name, an initial block count, a previous block count, a current block count, a linearity value, the first allocated memory block address, and contents of the first allocated memory block.

45. The apparatus of claim 41 further comprising:

a means for searching the first allocated memory block for a reference to at least one of the plurality allocated memory blocks; and

a means for storing the first allocated memory block address in a contingency chain corresponding to the at least one of the plurality allocated memory blocks when a reference to the at least one of the plurality allocated memory blocks is found in the first allocated memory block.

46. The apparatus of claim 41 further comprising:

a means for examining a contingency chain corresponding to one of the plurality of allocated memory blocks to determine whether any of the plurality of allocated memory blocks references the one of the plurality of allocated memory.

47. The apparatus of claim 41 further comprising:

a means for forming a contingency chain for each of the plurality of allocated memory blocks, wherein each contingency chain is indexed by an allocated memory block address of the corresponding each of the plurality of allocated memory blocks.